Big Data
Volume 4 Number 2, 2016

DOI: 10.1089/big.2016.0007

ORIGINAL ARTICLE

A Framework for Considering Comprehensibility in Modeling

Michael Gleicher*

Abstract

Comprehensibility in modeling is the ability of stakeholders to understand relevant aspects of the modeling process. In this article, we provide a framework to help guide exploration of the space of comprehensibility challenges. We consider facets organized around key questions: Who is comprehending? Why are they trying to comprehend? Where in the process are they trying to comprehend? How can we help them comprehend? How do we measure their comprehension? With each facet we consider the broad range of options. We discuss why taking a broad view of comprehensibility in modeling is useful in identifying challenges and opportunities for solutions.

Key words: data analysis; human-computer interaction; visualization; visual analytics; machine learning; statistical modeling

Introduction

Data-driven mathematical models—broadly, abstractions of data—have had a profound impact on a wide range of problems: They may predict unseen situations, provide compact descriptions of large data, permit inferences about populations based on samples, classify and organize data, or allow generation of new synthetic examples. Data-centric modeling is central to many "big-data" applications. Its impact has been enabled by considerable progress in the techniques for all stages of the modeling process: Ever-growing datasets are used as input to sophisticated computational implementations to construct complex models that are subjected to validation and human interpretation. The combination of statistics, machine learning, distributed systems, and data management has led to impressive results. Continued development in these fields, coupled with ever more available data sets, will, undoubtedly, lead to even more impressive systems that apply complex methods to address important problems.

However, even as data-centric modeling becomes more capable, its potential for truly widespread adoption and impact will be limited by human-centric challenges, for the increasing sophistication of the constructed models will also make them increasingly difficult to understand. The ubiquity and potential power of these systems will raise the importance and value of comprehensibility, which we define as *the ability of the various stakeholders to understand relevant aspects of the modeling process.*

The goal of this article is to provide a framework for considering comprehensibility in modeling to aid in identifying challenges and opportunities. The comprehensibility problem encompasses a wide range of scenarios such as domain scientists trying to discern meaning from their models, machine-learning developers trying to tune their algorithms, and the general public trying to decide whether to trust a newspaper's prediction. A broad definition of comprehensibility admits a wide range of potential challenges to be addressed.

A broad view of "*the* comprehensibility in modeling problem" is valuable, because issues and opportunities can arise in so many ways. A broad view not only may allow for the identification of an unexpected problem but also might suggest solutions that come from very different places. A pitfall with such breadth is that it requires some structure to organize the range of elements and to help see them in ways that expose their similarities. Therefore, this article takes the important first step of proposing a framework for considering the range of challenges and opportunities.

Department of Computer Sciences, University of Wisconsin-Madison, Madison, Wisconsin.

*Address correspondence to: *Michael Gleicher, Department of Computer Sciences, University of Wisconsin-Madison, 1210 West Dayton St., Madison, WI 53706*, E-mail: gleicher@cs.wisc.edu

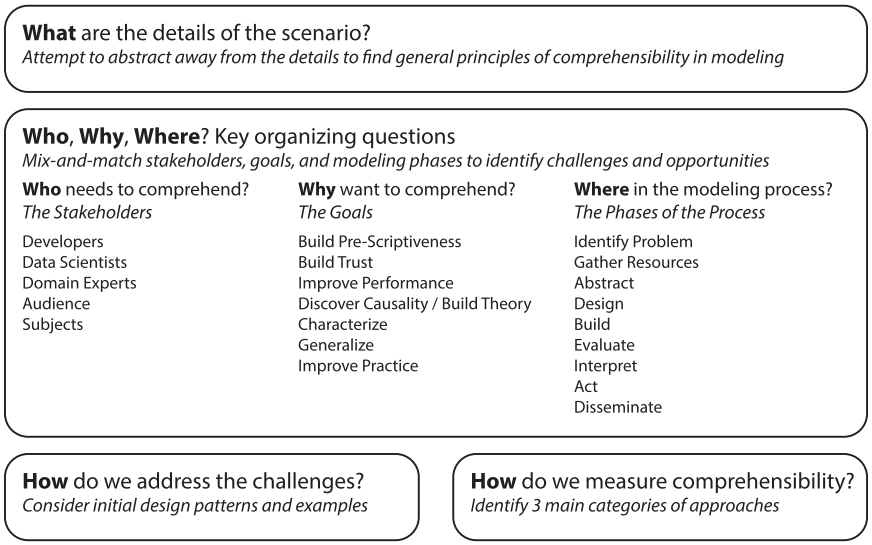

FIG. 1. A summary of the framework proposed in this article. The specific lists for each question are initial organizations to show the broad range of aspects to consider.

The proposed framework is summarized in Figure 1. It embraces the breadth of what comprehensibility in modeling involves with a multi-faceted space with many dimensions to be considered. Such a multi-faceted categorization strategy parallels some of the strategies developed for visualization (e.g., Schulz et al.[1]). The facets are chosen to view the modeling problem abstractly—without the specifics of a scenario. Each facet poses a question and provides a classification that can help group problems (and potential solutions) together: *Who* is comprehending? *Why* are they trying to comprehend? *Where* in the process are they trying to comprehend? *How* can we help them comprehend? *How* do we measure their comprehension? These questions are designed to help us abstract away from another question: *What* are the specifics of the scenario?

The key message of this article is to advocate for a broad consideration of the problem of comprehensibility in modeling. There are two main premises that are part of this. The first is that comprehensibility is an important consideration in modeling. The second is that the comprehensibility problem is best considered in a broad and multi-faceted manner. This involves not only multiple facets (or views) but also a consideration of the breadth of options that each encompasses. Such an exploration purposefully encourages a holistic view of modeling as a process involving many phases, many stakeholders, many potential goals, and (therefore) many potentials for comprehensibility problems and solutions.

A secondary aspect of this article is to advocate more broadly for the considerations of human-centric aspects of modeling. This article attempts to show that human-centric approaches, such as data visualization, have a wide variety of potential roles in successfully applying data to problems. This article does not attempt to provide a survey of prior successes, but rather to provide a framework for considering comprehensibility in a broad, multi-faceted manner.

Tradeoffs in comprehensibility

In modeling, there are many goals. For most scenarios, accuracy is a primary concern, whether it be how well the output of a predictive model matches reality or how well a descriptive model summarizes the underlying data. However, accuracy is not the only concern in modeling: Efficiency, generalizability, robustness, conciseness, verifiability, and self-consistency are some of the many other qualities that model designers and analysts must consider. These properties often form tradeoffs between one another: More complex algorithms may provide higher accuracy but take longer to build; models that use fewer variables may provide lower accuracy but be less susceptible to over-fitting. The proper balance in such tradeoffs depends on the context and needs.

Comprehensibility provides yet another potentially important goal. In different scenarios, its importance relative to other goals (accuracy, robustness, etc.) may vary. Tradeoffs exist. Indeed, the tradeoff between accuracy and comprehensibility is most often raised as an argument against comprehensibility. The tradeoff is often touted as inevitable (e.g., Huysmans et al.,[2] who "observe the inherent duality between comprehensibility and accuracy"), usually with a simple argument: More complicated models are capable of representing a richer space of functions and are, therefore, capable of better representing the underlying phenomena, although at the cost of understandability.

However, the tradeoff between comprehensibility and accuracy is not universal. In fact, comprehensibility may actually serve as a pathway to achieve other goals. In some cases (e.g., Stiglic et al.[3]), simpler (more comprehensible) models may improve accuracy. Other ways to achieve improved performance via comprehensibility are more common. For example, a tool for understanding what is happening inside a complex model can provide insights that suggest improvements (see Zeiler and Fergus[4] for a compelling case study).

The potential for tradeoffs with other goals provides an important motivation for understanding comprehensibility in a broad way. When tradeoffs do arise, the challenge of quantifying comprehensibility often makes sacrifices difficult: When one cannot measure how much comprehensibility is being gained for a (more easily measurable) loss in accuracy or computation speed, such a tradeoff can be difficult to justify.

Throughout this article, we consider making "interventions" to better meet comprehensibility goals. The term is meant to imply a choice made to improve comprehensibility. It might be an action to change an aspect of an existing process, or it could also be that the choice of a standard modeling approach in a situation already achieves good comprehensibility (either by design or by chance). An intervention may be something that trades some other property for comprehensibility, but it also might be a way to achieve comprehensibility without negatively impacting other needs.

A Framework for Comprehensibility

The broad definition of comprehensibility in modeling admits a wide space of scenarios. To help organize this diversity, the framework considers a set of basic questions. It suggests three questions to consider first: "*Who* is comprehending?" "*Why* are they trying to comprehend?" "*Where* in the modeling process are they trying to comprehend?" Considering these questions helps frame the problem so that we can better answer the two key *how* questions: "How do we help them?" and "How do we know that we have helped them?" (i.e., "How do we measure comprehensibility?") All of these questions are intentionally abstracted from the specifics of the scenario, which are loosely grouped in the question "What are they trying to comprehend?"

The idea of using the basic question words to create facets is owed to its famous application in journalism, where it is immortalized as "the five Ws." This was recently applied for analyzing data visualization tasks by Schulz et al.,[1] and a similar set of questions is used as a framework for visualization design by Munzner.[5] As in most visualization taxonomies (see Brehmer and Munzner[6] for a recent survey), they are focused on building visualization solutions to data interpretation problems, whereas the questions as posed here are meant to help identify what those problems are and what the range of solutions might be (beyond just visualization).

Prior work in the areas of machine learning and modeling has taken a more narrow view on comprehensibility. The machine-learning literature more often uses the term "interpretability" and almost always refers to understanding the learned model. For example, Craven[7(p.4)] defines comprehensibility based on the representation of the model: "does the learning algorithm encode its model in such a way that it may be inspected and understood by humans." This definition narrows the focus to the model itself, which is only one phase of the larger modeling process (see Section "Where: The Analysis Pipeline and Its Opportunities"). By considering more aspects of modeling, we can identify a broader range of problems and solutions. In this article, we use the term "comprehensibility" rather than "interpretability" to avoid the latter's connotation of focusing on interpreting the model itself.

What: details of the scenario

Our goal is to provide a general framework for considering comprehensibility in modeling that will apply across a wide variety of scenarios, and for allowing us to develop generalizable approaches and methods to address common problems.

There is a strong tradition of trying to abstract data analysis problems. For data visualization, there has been considerable effort to develop abstract ways to consider user tasks (see Refs.[1,6] for recent examples with extensive surveys). The provided taxonomies and categorizations can be helpful in being able to match

solutions to problems, to group-related solutions, to allow for comparisons, etc. However, they are too focused to provide for the broad view of the modeling process. Addressing comprehensibility in a modeling process may create a range of potential problems that could be addressed by many different types of interventions (including visualizations).

Any specific scenario of modeling involves a number of details: in some specific application or domain, with some specific type of data, for some specific model type built with some algorithm, etc. Choosing which of these details not to consider in building a framework is important. For example, a logical first question to ask in modeling is "*What* is being modeled?" Although the thing being modeled is certainly important to modeling, many of the issues of comprehensibility apply regardless of whether we are modeling collections of literary documents, protein molecules, or financial trends.

Who: the stakeholders

Considering comprehensibility implies that there must be someone to do the comprehending. There are many potential stakeholders in a predictive modeling application. Understanding this range of stakeholders is important. Each may have different needs for comprehension in the process and different abilities. The stakeholders in a prediction application may include:

Developers: people building tools and methods meant to work across a range of problems—for example, a researcher inventing new algorithms or implementations.

Data scientists: people who do the model building. This role involves using the general tools and methods in a way that is somewhat detached from the domain.

Domain experts: people who "have the problem." They often commission the model building and consume the results. In fields that develop general methods, such as visualization or statistics, such people are often referred to as "domain collaborators." The key distinction is that these stakeholders are primarily interested in the topic of the data, not the general problem of working with data.

Audience: people who ultimately get the results—for example, the audience of a scientist's paper or a journalist's article. The definition of audience is clouded by the fact that many different stakeholders may have "audiences," and in a sense, downstream stakeholders may be the audience of upstream ones (e.g., a scientist is the audience of a tool builder).

Subjects: people who are affected by the model but will not work directly with it. This includes, for example, patients who may receive improved care, because medical practitioners (audience) have learned from researchers' (domain experts) predictive results, or from consumers whose actions are being modeled in marketing research.

In practice, there might not be clean separation between roles. One individual might have multiple roles in the process.

Different stakeholders may have different requirements. However, each category of stakeholder can potentially have multiple kinds of goals (see Why: Goals for Comprehensibility section) and concerns about multiple aspects of the modeling process that they are interested in (see Where: The Analysis Pipeline and Its Opportunities section).

Different stakeholders may have different abilities. One might expect statistical sophistication among data scientists, but not necessarily in a general audience such as in an article for the popular press. Within each category of stakeholder, there is the potential for a range of expertise, skills, abilities, and motivations. Novice developers or domain scientists may have different needs, and be served by different tools, than experts.

Although problems (and solutions) are often specific to certain stakeholders, the issues that one group may have, and the solutions designed for them, may have implications for others. Often, interventions designed with one group in mind end up helping other groups as well (see Validation Experiment Visualization section for an example). In addition to considering "who is comprehending," it can be useful to think about "who else."

Another potential distinction is between model builders and model users. This distinction is often orthogonal to the stakeholder classification listed earlier. The roles of builder and user may be held by many of the different stakeholder types, and often a single stakeholder may take on both builder and user roles. These two roles also provide ways to think about comprehensibility, as there are comprehensibility challenges in each.

In future work, a more complete understanding of the range of stakeholders in the modeling process will be valuable, because it will better enable an understanding of the range of needs, and for the identification of needs and the design of approaches that address them.

Why: goals for comprehensibility

Comprehension of a model is rarely an end unto itself: Comprehensibility is typically desired, because it helps

in achieving modeling goals. Being able to identify the comprehensibility goal is important in defining a comprehensibility problem, and in seeing whether a solution is appropriate. If there is no reason for wanting comprehensibility, there is little reason to seek it, and less reason to make tradeoffs that sacrifice other goals.

An understanding of potential goals is important. It can help identify and articulate needs, assess tradeoffs, and provide the basis for assessment. It can help articulate the value of solutions and create an open-mindedness toward solutions by de-coupling them from the real objectives. Craven[7] provides a list of potential goals as an argument for the value of comprehensibility. Although he considers a narrow definition of the problem (restricting it to model representations), his goals (Validation, Discovery, Explanation, Improving Accuracy, Refinement) cover a similar range to the list proposed next.

However, development of an abstract vocabulary of goals is challenging. To the extent possible, such a list should be abstracted from the other facets and aspects of scenarios, complete (in that it covers the range of goals), and non-redundant (so that specific goals fall into few categories). Although this initial list of abstract goals may not achieve these aims, it provides a starting point for discussion.

Build pre-scriptiveness: understanding how to make use of the model and results. Pre-scriptiveness is the quality of a model that allows its user to do something with a result, for example, its ability to inform action. For example, a weather forecast can be used to choose appropriate clothing. Although some "prescriptive models" explicitly consider decision making,[8] decision support often relies on interpreting predictive or descriptive models.

Build trust: understanding a prediction to appropriately trust it, or a predictive process to trust in its ability to make predictions (and to trust those predictions).

Improve performance: Understanding can lead to improvements in many of the other desired properties (e.g., accuracy and efficiency). Understanding can drive iterative refinement that is applied to improve predictive accuracy, efficiency, robustness, and even comprehensibility.

Discover causality/build theory: Although the modeling process typically has its main goals of modeling results (e.g., making predictions), it can often have the side effect of shedding light on the underlying process that is being modeled. Although a statistical model usually uncovers correlations, not really identifying causality, it can be a useful starting point for theory building, or even an empirical approach toward testing theory. Craven[7] differentiates within the goal of theory building, separating the goals of discovery (finding unknown relationships) and refinement (refining approximately correct domain theories).

Characterize: understanding what the model can do and where it can be applied. For example, even if models do not have an explicit uncertainty characterization for their predictions or an explicit characterization of their operating regions, a deeper understanding of various aspects of the model can help create them.

Generalize: Understanding can help extend modeling work to situations beyond its original goals, for example, to see that methods might apply in other applications.

Improve Practice: Understanding the success (or failures) in one model, modeling application, or modeling process can help prepare the stakeholder for future applications. The pedagogical value of understanding a particular model can go beyond that specific instance and help the learner better understand modeling in general.

Many fields, including social and management sciences, consider how people and organizations achieve similar goals in other situations.* This may provide a rich source of potential ideas, both for characterizing the range of tasks and for inspiring approaches to support them.

Although it is tempting to associate individual goals with specific stakeholders and aspects of the modeling process (Who: The Stakeholders and Where: The Analysis Pipeline and Its Opportunities sections), an important observation is that any goal might be had by a range of stakeholders and might be addressed in a wide variety of ways.

Where: the analysis pipeline and its opportunities

Modeling is a process. The process begins before data gathering and wrangling, moves through the phases of mathematical model building and validation, and does not end until at least the users have had a chance to act on the predictions made. And of course, it may not end there, as reflection, reuse, and revision make

*I thank the anonymous reviewer who suggested this connection and provided some initial pointers into the literature.

the process potentially cyclic. Comprehensibility can be a factor in any of these phases in the process. Thus, comprehensibility can mean many different things.

Identification of the phases of the modeling process has been done in many ways. Often, fields (such as developers of statistical techniques, or data wranglers) focus on stages that they are most involved in. For example, the National Institute of Standards and Technology's Handbook of Engineering Statistics describes modeling as a three-phase process (model selection, fitting, and validation).[9(S.4.4.1)] The mathematics education literature sometimes provides a more comprehensive enumeration of the phases. For example, Anhalt and Cortez[10] provide a list of six steps (Analyze Problem; Develop and Formulate Model; Compute Solution; Interpret/Draw Conclusions; Validate Conclusions; Report). It is not a coincidence that this more comprehensive view comes from the community that is focused on how people learn (and understand) the modeling process: Stakeholders are involved in many different activities, and all of these are likely to be important to learn (and, by extension, to understand).

The data-mining community has made many attempts at trying to define a process model that identifies the common steps (see Refs.[11,12] for surveys). These models often include a broader range of steps, emphasizing that early aspects (e.g., planning) and late aspects (e.g., usage) are often critical to success of the overall process. Although these different process models divide the steps differently, they share a common range of steps. The common CRISP-DM model breaks the process into six steps: business (domain-problem) understanding, data understanding, data preparation, modeling, evaluation, and deployment.

Taking a fully broad view of data-centric modeling exposes a very broad range of activities in building and using a model, leading to a longer list of phases.

Identify the problem.

Gather resources to address the problem. This includes data collection.

Abstract the problem.

Design the model. This encompasses the "formulation" phase where the choices about the model, such as the methods to be used, are made.

Build the model. This is the computation (e.g., fitting) of the specific model.

Evaluate the model. This includes validation, and whatever other kinds of testing and checking might be done to assess the quality of the model.

Interpret the results.

Act on the results.

Disseminate the results.

The length of the list emphasizes the diversity and breadth of tasks. Although the specifics of how the process is divided is more a matter of naming, the important lesson is that modeling involves a range of steps, from planning through dissemination, and that any stage may be the source of a comprehensibility problem, or an opportunity for an intervention (often to address an issue in a different phase).

A different way to think about the modeling pipeline is to consider the objects used by the processes, rather than the tasks (this was my approach in Ref.[13]). Each provides a tangible thing that someone might need to comprehend.

Inputs and resources: Does the stakeholder understand the data, assumptions, and initial questions being used to build the model? For example, we might have a collection of training data or a causal theory that leads to a mathematical model.

Methods: Does the stakeholder understand the method used to build the model from the data (e.g., the algorithms)? To illustrate, in a linear support vector machine (SVM) learning approach that is used to convert training data into a classification model, the method comprises both the general method (i.e., linear SVM) and the specific implementation of that method (e.g., a particular machine-learning software package). In creating a model from some underlying principles, the method might be a symbolic derivation.

Model: Does the stakeholder understand the model used to make predictions? For example, this often takes the form of the specific equations that compute predictions. For the SVM example, the model is the set of coefficients (the linear equation) created. As mentioned earlier, the notion of interpretability in machine learning tends to focus on this phase.

Outputs: Does the stakeholder understand specific predictions made by the model? The output is the prediction computed for a specific condition. It may include more than the prediction itself. For example, predictions often include confidence values, strength

scores, uncertainty quantification, or sensitivity analyses. Output comprehension includes both the semantics of the result (e.g., what does the prediction of a specific value for the predicted variable mean) and the mathematical meaning of the prediction. For a common example, a weather prediction ("20% chance of rain tomorrow") may say little about any specific time in the day, the amount of rain that might fall, or whether it will be sunny.

Experiments: Does the stakeholder understand the results of experiments run on the model? Experiments include predictive performance evaluation (e.g., cross-fold validation or holdouts) and efficiency tests (e.g., profiling). The results of experiments can include not only the outputs over a set of cases (often with known ground truth) but also various statistical information.

Presentations: Does the stakeholder understand the artifacts used to present the results? This might include things such as writing that describes the results, figures, videos, spoken presentations (live or recorded), etc.

How (1): pathways to comprehensibility (how do we help?)

The big question for visualization research and practice is "How can I intervene to address comprehensibility in modeling?" The previous questions help us identify problems (or opportunities) and categories that solutions might address. They are intentionally abstracted from what the solutions actually do to address the problems with a goal of being open to a range of potential approaches to interventions.

Several research areas, including visualization, machine learning, and various domain sciences, provide examples of interventions that seek to address comprehensibility issues. An even broader range of work may address such problems implicitly, for example, by providing better methods.

From examining a selection of representative examples, some general strategies for comprehensibility interventions begin to emerge. This initial list of "design patterns for comprehensibility" is provided to show the potential for such generalized strategies and to give some hint at the range of potential solutions that are likely to be developed. Cataloging and characterizing the range of comprehensibility interventions, in addition to creating new ones, will be important future work.

Aggressive regularization.

Many model-building techniques explicitly encode a tradeoff between how well the model fits the training data and some measure of the representation of the answer. In optimization parlance, this is known as *regularization*. Regularization augments a primary optimization objective (e.g., how well a learned model fits its training data) with a secondary objective (e.g., the magnitude of the coefficients, the number of coefficients, or the depth of the decision tree). By penalizing answers whose representations are less desirable (by the secondary metric), better overall answers may be achieved even if the optimal value of the primary objective is not found. This secondary objective is known as a regularization term, and the amount that it is considered (relative to the primary objective) is often a tunable parameter that we might consider the degree of regularization.

Many machine-learning and statistical modeling algorithms provide a parameter to tune the degree of regularization. Such parameters are often used as a tool to control over-fitting. Since the metric used for regularization is often a proxy metric for comprehensibility, control of the amount of regularization can be a way to control the accuracy versus comprehensibility tradeoff. Aggressive regularization, using large amounts to obtain simpler answers, is therefore a strategy for improving comprehensibility. A common form is to use algorithms where the regularization terms prefer sparsity (e.g., $L1$ norms in SVMs, LASSO, or logistic regressions).

Aggressive regularization can be straightforward to implement, as regularized modeling techniques are well established and provide a very direct control over the accuracy versus simplicity tradeoff. However, this strategy has a number of pitfalls. Foremost, it relegates comprehensibility to a tradeoff, rather than trying to find ways to achieve comprehensibility along with other goals. Second, it assumes that the regularization metric is a good measure of comprehensibility—when in practice, it is (at best) a proxy metric. Related, it is limited to proxy metrics that have been considered important by algorithm developers. Algorithms for achieving sparsity (minimizing the number of variables) are common in machine learning and statistics, whereas algorithms for other potential simplicity objectives are less common (see Refs.[14,15] for algorithms that also try to find simple coefficient values). Also, existing algorithms are often not designed for aggressive regularization; instead, they are being tuned for smaller amounts of regularization for other purposes (like stabilizing the computation). For instance, in our examples with $L1$-regularized SVM,[14] we found that

standard SVM solvers could not achieve extreme regularized solutions (finding optimal sets of a very small number of variables).

Aggressive regularization is an attractive intervention for addressing comprehensibility concerns, as it provides control over the tradeoff between accuracy and comprehensibility (or, to be more precise, properties believed to correlate with comprehensibility). Key future work includes developing more direct connections from regularization terms to comprehensibility and understanding how stakeholders can utilize control over tradeoffs to address their needs.

Model reprojections. Once a complex and successful model is built, that model can be used as a basis to build other models with simpler forms. An example is to first build a neural network classifier for a problem, and then to use this model to build a classifier using a different modeling strategy that is believed to be easier to comprehend, such as a rule-based or decision-tree-based classifier. Although such approximation often results in poorer performance, it does not need to do so. Potentially, information gained from the process of building, using, and exploring the initial model can allow for constructing simpler models that could not have been constructed without these insights.

A canonical form of model reprojection is *rule extraction:* building decision rule classifiers from other kinds of classifiers that are considered more complex to interpret. Rule extraction has been explored extensively in the machine-learning community (see Huysmans et al.[2] for an extensive survey). However, rule extraction generally assumes that decision rules are desirable from a comprehensibility point of view.

Because the simpler models often have worse performance, it may make sense to use the original (complex) models for applications where accuracy matters, and reserve the simplified models for situations where interpretation is more central. Johansson et al.[16] suggests constructing an ensemble of models at various levels of tradeoff, so that appropriate ones can be used.

Another reprojection strategy is to use aspects of a complex model as inputs to a second modeling process. For example, a first model construction may identify a set of features that are used in the training process for a more interpretable form of classifier. A concrete example is presented by Stiglic et al.,[3] who use interaction analysis to identify new features that involve multiple input features and then use these features to derive simple trees. More commonly, complex neural nets can be interrogated to understand how their initial layers serve as feature finding filters—either as recognizable filters[17] or as specialized detectors.[18] Indeed, some network learning approaches are specifically designed to discover structure in data.[19]

Reprojection is an attractive strategy for comprehensibility interventions, because it allows using a model that is successful but addresses comprehensibility concerns. It affords the potential for designing new modeling methods specifically for their comprehensibility properties, with less concern for their ability to achieve accuracy goals. Important future work includes both how model type influences comprehensibility and how to make the inherent tradeoffs transparent to stakeholders.

Visualizing complexity. The previous two approaches focus on finding simpler models. An alternative is to simply accept the richness of a given model and instead find ways to better present and explore it. Such a strategy might be employed with a variety of complex objects in the modeling process. For example, specific approaches might visualize the internal representations of a deep neural network or enable visual exploration of complex validation experiments.

With a complex model, there is a dichotomy of strategies for visualization: visualizing the internal structure of the model versus presenting the relationships encoded by the model. For example, Tzeng and Ma[20] provide visualizations of neural network architectures (internals) whereas Cortez and Embrechts[21] derive visualizable representations of the relationships (sensitivities) between model inputs and outputs.

For models with a very complex internal structure, direct visualization may not be possible. For example, with neural networks, visualizing the network structure may be possible for simpler networks,[20] or connections between input notes and data patterns can be seen by qualitative inspection.[17] However, as the networks grow larger and deeper, interpretation requires some degree of translation. To "look inside the mind" of a complex neural net (to use the terms of Hinton et al.[22]), techniques may be needed to construct more interpretable representations. For example, Hinton et al.[22] introduced a strategy of using sampling to infer what kinds of inputs would be required to activate different internal components. This strategy has the advantage of relating internal structures to (hypothetical) input examples, which are more likely to be

familiar than activation patterns. Other methods for approximating these relationships have been developed (e.g., Lee et al.[23]), and the general approach of sampling to look inside a network has been empirically shown to be robust.[17] Similarly, extant methods for interpreting random forest classifiers focus on input and output relationships, rather than on model internals.[24]

Chuang and Socher[25] provide an example where examination of complex internal structures is successful. They allow for the exploration of collections of examples of how natural language methods represent sentences. This exploration can be used for performance improvement, as it allows a developer to identify what phase of processing errors occur, and to identify opportunities to engineer new features.

A different approach to visualizing complexity is to build visual and interactive techniques that connect complex models to the users' needs and workflows. An example of such an approach is the Serendip system[26] for interaction with topic models of text corpora. The system does not hide the complex internals of the models used, but it does provide a connection between relevant elements and specific user tasks.

Effective communication of complexity, through visual representations and interaction, is attractive, because it makes direct use of models that may be desired for other reasons (such as performance). As visualization and visual analytics methods mature to provide approaches that scale to increasing complexity, opportunities to apply this work to complex modeling applications will also improve. Important future work includes understanding how to apply general capabilities for presenting complex data to specific comprehensibility challenges.

Interaction around model construction. Muhlbacher et al.[27] surveys many ways that systems can enable users to interact with modeling algorithms where the algorithm itself is treated as a black box. Their work characterizes a space of opportunities for interactions around a model-building step, such as changing inputs, altering parameters, or examining intermediate results. They do not focus on any particular task—in particular, improving comprehensibility. However, many of their strategies achieve modeling goals by a user's extant comprehension of various aspects of the modeling process.

The space of interactions mapped out by Muhlbacher et al.[27] suggests a wide range of potential strategies for working with models. Many have potential applications for comprehensibility interventions. For example, they suggest "user subsetting" as an approach to achieve faster iterations in the modeling cycle. This approach might also be explored as a strategy to comprehensibility, as models built on subsets may be easier to understand, and the understanding built on the reduced models may be scaled to more complex cases.

Important future work includes understanding how the range of modeling interactions might serve the range of comprehensibility needs.

How (2): metrics (how do we assess?)

In discussing comprehensibility, an inevitable question is "How do we measure it?" If comprehensibility is perceived to be a problem, then there is a sense of not having enough of it. If a comprehensibility intervention is proposed, we may want to assess how much more understanding is achieved. When different modeling approaches are compared, we may want to know which provides more comprehensibility. Objective measurement can be particularly important when tradeoffs are involved. Quantifying a loss in accuracy or efficiency can be easy, whereas it is much harder to quantify the comprehensibility gained in its place. How much comprehensibility is gained by reducing a model by two variables? Is this worth a 2% decrease in accuracy?

Unfortunately, assessing comprehensibility seems quite challenging. The breadth of what comprehensibility might mean, the human-centric and often subjective nature of it, and the relative lack of attention paid to it complicate the development of mechanisms for comprehensibility assessment. To date, our toolbox of approaches is not very well developed. However, in the limited existing literature, three main strategies seem to be emerging: direct measurement of comprehension, direct measurement of goals, and proxy metrics.

Direct measurement. A direct measurement of comprehensibility tries to assess the stakeholders' understanding. Such measurements are hard, because they seek to measure what is happening for the stakeholder internally and the metrics are often not well defined. However, direct measurement is possible, for example, by asking questions (either objective or subjective), or even through biometric assessment.

Assessment strategies from the learning sciences and human factors may provide a source of tools for direct measurement of comprehensibility. Already, measurements such as cognitive load and mental effort have been adapted to measure model quality in other domains.[28,29]

Goal-driven metrics. A goal-driven metric of comprehensibility measures the effects of comprehensibility, ideally in assessing how well its goals are being achieved. For example, if comprehensibility is sought to help improve accuracy of a model, then the accuracy gains provide a yardstick. Although such measurements are attractive because they get at the real reasons for wanting comprehensibility, they cannot always attribute the results or provide ways to see other collateral costs and benefits.

Goal-driven metrics are the most obvious for easily quantifiable goals, such as accuracy or efficiency. However, there is the potential to turn other goals into metrics as well. For example, insight quantification approaches[30,31] could measure how well a model supports hypothesis formulation (building theory).

Proxy metrics. The common metrics for comprehensibility are measures of the underlying model itself: for example, the type of model (e.g., a classifier might be a decision tree, an SVM, or a neural net) and measurements of the model (e.g., the depth of a decision tree, or the number of non-linear coefficients of a linear function). Such measurements do not measure comprehensibility *per se*, but rather they measure model properties that are believed to correlate with comprehensibility (e.g., decision trees are believed to be easier to understand than SVMs, and shallow trees are believed to be easier to understand than deeper ones).

Proxy metrics are attractive, because they are simple, quantitative, and are often well studied for other reasons such as to reduce overfitting. For example, the modeling communities (e.g., machine learning and statistics) have extensively explored the sparsity of models (having few non-zero coefficients). Although they tout comprehensibility as one of the benefits of sparsity, they also seek sparsity for other reasons such as better generalizability and more efficient computation. This, in turn, makes proxy modeling convenient for comprehensibility: The existence of good methods for creating and using models that optimize the proxy metrics means that models can be created with these properties should we desire them for comprehensibility purposes (see Aggressive Regularization section).

There are many pitfalls to proxy metrics. Foremost, their impact on comprehensibility is neither well understood nor quantified. Although the basic principles behind the metrics are often intuitive, a more sound understanding of them may still be helpful. For example, the oft-cited metric of sparsity is probably not incorrect: Equations with fewer variables are probably easier to understand than those with more variables. However, empirical evidence, or richer cognitive rationale, for the impact on sparsity is lacking. Quantification of the benefits of three variables instead of four is still challenging. Similarly, Craven[7] suggests *syntactic complexity* as a metric for measuring the comprehensibility of rule-based models. This metric is defined based on other proxy metrics (e.g., tree depth and number of feature references), and it is supported based on cognitive arguments that smaller models are preferable.

Another issue with proxy metrics is that the most convenient ones are often explored to the exclusion of others. Although sparsity is often discussed, others are rarely considered. However, in terms of interpreting a model, other aspects of the terms may be important. For example, having simple values for the coefficients (e.g., small integers) and choosing familiar variables (as opposed to less familiar ones) may be as important as having small numbers of variables (see Gleicher[14] for a discussion).

Examples: The Value of the Broad View

The previous sections introduced a framework for considering comprehensibility in modeling. The framework attempted to be broad: considering a number of facets of problems, and taking a holistic notion of the range of modeling in each. This section examines some examples from my own work to show the benefits of such a broad view.

Validation experiment visualization

By considering the whole modeling process, we can identify a broader range of opportunities to address comprehensibility issues, and these interventions may help achieve a range of goals. As an example, consider the validation experiments used to test and assess models. Although such experiments are an essential part of a modeling application, and are often complex and rigorous, the analysis of their results is often limited. For classification (machine learning) applications, the results are typically reported as a single number (e.g., accuracy, precision, recall, F1, etc.) or as a small set of these metrics.

In Sarikaya et al.,[32] we introduced a visual analytics approach for examining the results of protein surface classifiers, as shown in Figure 2. We sought to aid our structural bioininformatics domain collaborators (*who:* data scientists, model builders) in improving the

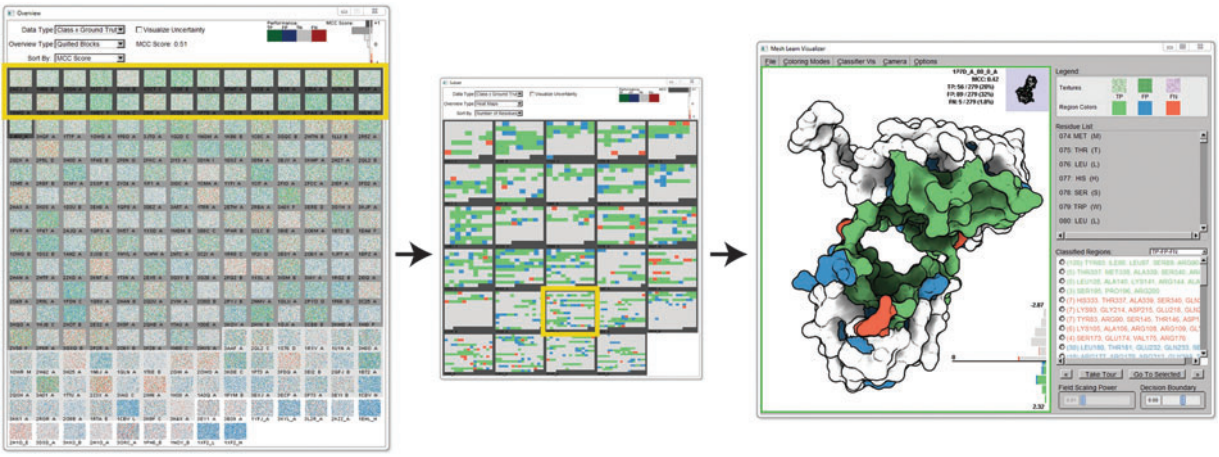

FIG. 2. Visualization of a validation experiment for a DNA-binding surface classifier that allows exploration of classification results. The corpus overview (left) is configured to display each molecule in the test set as a quilted glyph and orders these glyphs by classifier performance to show how performance varies over the molecules. Those proteins that appear more green have more true positive classifications, whereas those molecular that appear more red or blue have more misclassifications (false negatives and false positives, respectively). Selected molecules (left, yellow box) are visualized as heatmaps in a subset view (middle) and ordered by molecule size to help localize the positions of errors relative to correct answers. The detailed view (right) shows a selected molecule to confirm that most errors (blue, red) are close to the correctly found binding site (green).

performance of a classification system that they had built (*why:* improve performance). Although this *who* and *why* are standard, we chose a non-standard *where:* the validation experiments. Because our collaborators were simply using summary statistics of the experiment, this phase seemed to be under-exploited. Although our assessment in this work was informal, the specific goal gave us a success criterion: if our collaborators were able to gain insights that helped them devise performance improvements. Our approach was to provide a system that allowed for an overview examination and a detailed exploration of the entirety of the results of an experiment, which is complex because it involves dozens of hundreds of molecules, each of which is a complex three-dimensional shape on which the specific classification examples were made (*how:* embrace and present complexity).

The surface classifier validation experiment viewer was able to achieve its initial goal of helping data scientists improve the classification performance of their system. It also served other roles. Our collaborators used the system to show their results to their audience (structural biologists interested in understanding how these molecules perform their function). By observing patterns in where classification failed, an audience in a talk was able to hypothesize about alternate (biochemical) mechanisms that might be involved in this subset of a situation (*who else:* audience, *why:* Discover Causality/Build Theory).

A lesson of this example is that comprehensibility challenges, and therefore opportunities, can occur in many phases of the modeling pipeline. By mixing and matching different stakeholders, goals, and pipeline phases, we can identify potential interventions. In this case, our intervention was specialized (a system for examining validation experiment results on molecular surface classification data), but it suggests a more general class of tools.

Gotz and Sun[33] also presents a system for visualizing model errors over a test corpus that is aimed at identifying problematic training examples, problematic features, and opportunities for specializing models to subsets of the operating region.

Repurposing classifiers

By considering comprehensibility as a first-class objective, we can re-use modeling technologies for different purposes. For example, classifiers are models that are

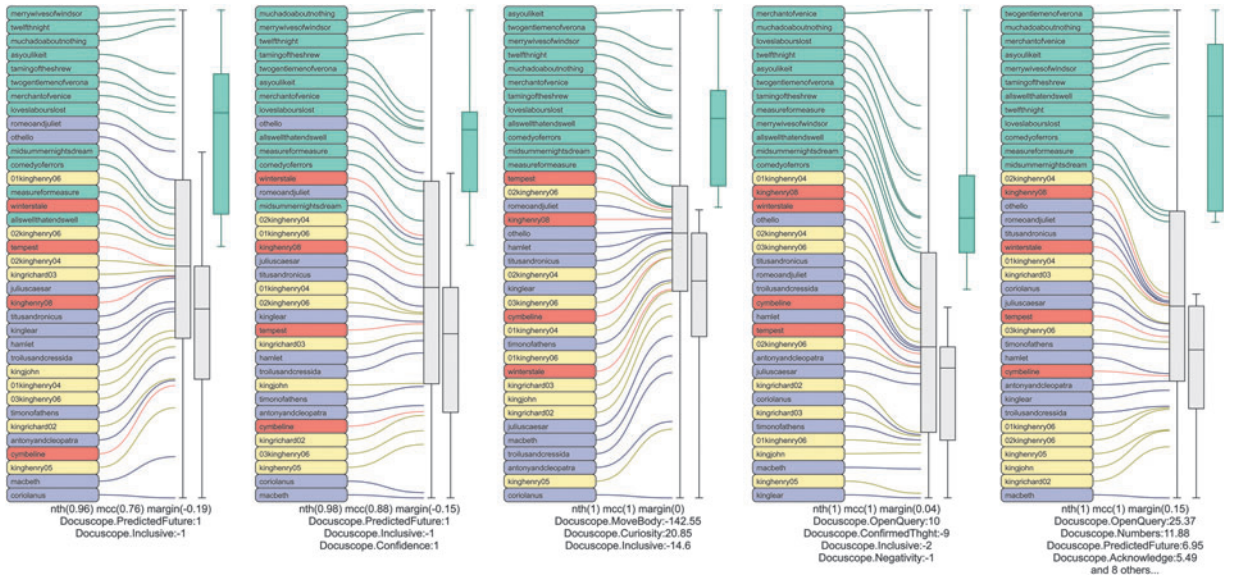

FIG. 3. Visualization of example *Explainers*, classifiers constructed with tradeoffs that emphasize comprehensibility concerns.[14] In this example, Shakepeare's 36 plays are measured with a set of 115 "Docuscope" features. Classifiers are constructed to identify the 12 comedies (green). Each column represents a linear SVM classifier, with the plays sorted according to their score. The leftmost classifier uses only two features with unit coefficients. It makes several mistakes (e.g., misclassifying the tragedies *Othello* and *Romeo and Juliet* as comedies), but the simplicity of the classifier makes it useful for building theory about how Shakespeare used the linguistic constructs in the different genres. In contrast, other classifiers may use more features and more complex weights to achieve better accuracy (and larger SVM margins), at the expense of how easy the functions are to comprehend. SVM, support vector machine.

typically used to classify—their standard use cases involve identifying unseen objects. However, when classifiers are constructed with comprehensibility in mind, they can be used for other purposes, such as helping domain experts build theories, generate hypotheses, and organize data.

In Gleicher,[14] we explore such an application. Our initial goal was to help literature scholars (*who:* domain expert) generate hypotheses and identify interesting examples for a detailed examination (*why:* Discover Causality/Build Theory). Specifically, they were interested in understanding the relationship between measurable features of documents (words that can be tagged as having certain rhetorical purposes) and higher-level features of the documents (such as its author, date, or genre). We sought to help them interpret the models that were being built (*where:* build model), with the idea that the internals of the model may expose connections, showing what a machine might use to classify the documents (that the scholars would classify using far different information). Unfortunately, most models were difficult to interpret, so we chose to introduce new modeling techniques that traded accuracy for comprehensibility (*how:* aggressive regularization).

We used the term *Explainer* for a classifier constructed with the purpose of being interpretable, at the expense of accuracy. An example is illustrated in Figure 3. The work showed many of the issues that must be addressed to realize such an approach. We observed that sparsity (limited numbers of variables) was valuable for interpretability, and other properties were valuable as well, including simple coefficients (small integers) and preferring familiar variables. We observed the importance of providing the user control over the tradeoffs between the various objectives (e.g., accuracy vs. sparsity), which required developing algorithms that supported the appropriate control. We observed the importance of methods for showing the resulting models so that a user could both assess its quality (in terms of accuracy) and interpret its structure. Although the initial explainers' paper[14] only provided initial

answers to these questions, it showed the viability of the approach by demonstrating methods for these various aspects.

Because the classifiers were simple, usually involving only two variables, they were easy to present to the (often skeptical) audience (*who else:* audience). Individual examples could be examined in detail, allowing the audience to learn about the process used (particularly, a specific choice of feature type), which helped establish trust in the approach. However, this required development of different ways to show the results, as audiences had different abilities and goals for interpreting presentations of the data.

Conclusion

There are many examples of work that identifies, or intervenes to address, issues of comprehensibility in modeling. These success stories are often specific: identifying a scenario with a specific domain, a specific modeling approach, a specific comprehensibility problem, and a specific intervention. A framework can help organize these examples around the various facets of the overall problem of comprehensibility in modeling. It can help identify commonalities to generalize as well as to identify unmet problems and unexploited opportunities.

This article has introduced a multi-faceted framework for considering comprehensibility in modeling. It suggests a variety of views (who, why, where, and how). Keeping a broad view of the range of answers to each question is valuable. Any stakeholder (from data source to audience) may have a comprehensibility issue. Any phase of the modeling process, from initial planning through final dissemination of results, may provide a challenge or an opportunity for intervention. Many goals may be helped by comprehensibility, and a range of approaches for interventions might help address issues.

The initial framework provided in this article may be extended by providing additional facets, or by better characterizing the range of possible answers within a facet. However, even in its current form, the framework can serve a role in the more important effort to identify comprehensibility issues, develop interventions that address them, and help modeling address even more challenging and important problems.

This article has made an assertion that human-centric considerations of comprehensibility are applicable across a very broad range of modeling, and the very broad range of stakeholders and tasks that use modeling. Admittedly, support for this argument has been provided with limited examples across a narrow range. Moving forward, the key future work is to develop our arsenal of interventions to address the broad range of comprehensibility concerns. This includes both collecting and characterizing the existing approaches, as well as developing new ones.

The key lesson from the presented framework is that comprehensibility in modeling has numerous facets, each of which covers a broad range. By taking this broad view, for example, considering the entire range of potential stakeholders or the range of steps in the modeling process, more challenges and opportunities for solutions can be identified.

Acknowledgments

This work was supported in part by NSF award 1162037, NIH award 5R01AI077376-07, and a grant from the Andrew Mellon Foundation. The author would like to thank the many people who had provided feedback for developing ideas, in particular, Remco Chang for his help throughout the process, and Ross Maciejewski, Eric Alexander, and Deidre Stuffer for their help in articulating the ideas in this article.

Author Disclosure Statement

No competing financial interests exist.

References

1. Schulz H-J, Nocke T, Heitzler M, Schumann H. A design space of visualization tasks. IEEE Trans Vis Comput Graphics. 2013;19:2366–2375.
2. Huysmans J, Baesens B, Vanthienen J. Using rule extraction to improve the comprehensibility of predictive models. SSRN 2006. Available at: http://dx.doi.org/10.2139/ssrn.961358
3. Stiglic G, Povalej Brzan P, Fijacko N, Wang F, Delibasic B, Kalousis A, Obradovic Z. Comprehensible predictive modeling using regularized logistic regression and comorbidity based features. PLoS One. 2015;10:e0144439.
4. Zeiler M, Fergus R. Visualizing and understanding convolutional networks. In Fleet D, Pajdla T, Schiele B, Tuytelaars T (Eds.): ECCV 2014, Volume 8689 of Lecture Notes in Computer Science, Cham: Springer International Publishing, 2014. pp. 818–833.
5. Munzner T. Visualization Analysis and Design. Boca Raton, FL, CRC Press, 2014.
6. Brehmer M, Munzner T. A multi-level typology of abstract visualization tasks. IEEE Trans Vis Comput Graphics. 2013;19:2376–2385.
7. Craven M. Extracting comprehensible models from trained neural Networks. Ph.D. Dissertation, University of Wisconsin–Madison, 1996.
8. Bertolucci J. Big data analytics: Descriptive vs. predictive vs. prescriptive. Information Week, December 2013.
9. NIST/SEMATECH. e-Handbook of statistical methods. Technical Report, 2012.
10. Anhalt CO, Cortez R. Mathematical modeling: A structured process. Math Teacher. 2015;108:446–452.
11. Marban O, Mariscal G, Segovia J. A data mining & knowledge discovery process model. In: Ponce J, Karahoc A (Eds.). Data Mining and Knowledge Discovery in Real Life Applications, Chapter 1. I-Tech Education and Publishing, 2009.
12. Kurgan LA, Musilek P. A survey of knowledge discovery and data mining process models. Knowledge Eng Rev. 2006;21:1–24.
13. Gleicher M. Position paper: Towards comprehensible predictive modeling. In: Visualization for Predictive Analytics Workshop, Paris, France, 2014.

14. Gleicher M. Explainers: Expert explorations with crafted projections. IEEE Trans Vis Comput Graphics. 2013;19:2042–2051.
15. Ertekin S, Rudin C. A Bayesian approach to learning scoring systems. Big Data. 2015;3:267–276.
16. Johansson U, Konig R, Niklasson L. Automatically balancing accuracy and comprehensibility in predictive modeling. In: 2005 7th International Conference on Information Fusion, volume 2, p. 7, 2005.
17. Erhan D, Bengio Y, Courville A, Vincent P. Visualizing higher-layer features of a deep network. Technical Report, Department IRO, University of Montreal, 2009.
18. Mohamed A-R, Hinton G, Penn G. Understanding how deep belief networks perform acoustic modelling. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 2012. pp. 4273–4276.
19. Hinton G, Osindero S, Welling M, Teh Y-W. Unsupervised discovery of nonlinear structure using contrastive backpropagation. Cogn Scie. 2006;30:725–731.
20. Tzeng F-Y, Ma K-L. Opening the black box—Data driven visualization of neural networks. In: VIS 05. IEEE Visualization. Minneapolis, MN, 2005. pp. 383–390.
21. Cortez P, Embrechts MJ. Opening black box data mining models using sensitivity analysis. In: 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), Paris, France, 2011. pp. 341–348.
22. Hinton GE, Osindero S, Teh Y-W. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18:1527–1554.
23. Lee H, Grosse R, Ranganath R, Ng AY. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning—ICML'09, New York, NY, 2009. pp. 1–8.
24. Golino HF, Gomes CMA. Visualizing random forest prediction results. Psychology. 2014;5:2084–2098.
25. Chuang J, Socher R. Interactive visualizations for deep learning. In: Visualization for Predictive Analytics Workshop, Paris, France, 2014.
26. Alexander E, Kohlmann J, Valenza R, Witmore M, Gleicher M. Serendip: Topic model-driven visual exploration of text corpora. In: 2014 IEEE Conference on Visual Analytics Science and Technology (VAST), Paris, France, 2014. pp. 173–182.
27. Muhlbacher T, Piringer H, Gratzl S, Sedlmair M, Streit M. Opening the black box: Strategies for increased user involvement in existing algorithm implementations. IEEE Trans Vis Comput Graphics. 2014;20:1643–1652.
28. Moody D. Cognitive load effects on end user understanding of conceptual models: An experimental analysis. In Benczúr A, Demetrovics J, Gottlob G (Eds.): Advances in Databases and Information Systems: LNCS 3255, volume 3255 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. pp. 129–143.
29. Zugal S, Pinggera J, Reijers H, Reichert M, Weber B. Making the case for measuring mental effort. In: Proceedings of the Second Edition of the International Workshop on Experiences and Empirical Studies in Software Modelling—EESSMod'12, New York, NY, 2012. p. 1.
30. North C. Toward measuring visualization insight. IEEE Comput Graph Appl. 2006;26:6–9.
31. Saraiya P, North C, Duca K. An insight-based methodology for evaluating bioinformatics visualizations. IEEE Trans Vis Comput Graphics. 2005;11:443–456.
32. Sarikaya A, Albers D, Mitchell J, Gleicher M. Visualizing validation of protein surface classifiers. Comput Graphics Forum. 2014;33:171–180.
33. Gotz D, Sun J. Visualizing accuracy to improve predictive model performance. In: Visualization for Predictive Analytics Workshop, Paris, France, 2014.

Cite this article as: Gleicher M (2016) A framework for considering comprehensibility in modeling. *Big Data* 4:2, 75–88, DOI: 10.1089/big.2016.0007.

Abbreviation Used

LASSO = least absolute shrinkage and selection operator
SVM = support vector machine